Amendments to the Specification

Applicants respectfully request that the Examiner enter the following amendment to the specification.

On page 8, please replace the paragraph beginning on line 2 with the following paragraph:

Figures 2a-2b illustrate the input data word re-alignment unit **102a** for generating the intermediate data words with the data bytes preceding and following the data byte insertion point of the current cycle (if any) repositioned appropriately within the intermediate data words, in further detail, in accordance with one embodiment. As alluded to earlier, the embodiment assumes the size of each data word processed in each cycle to be 64 bits (eight (8) bytes). Further, the data byte insertion position is denoted in an unconventional manner with the data byte insertion position "8" denoting that the insertion is to be made before the input data word of the current cycle, and "0" denoting that the insertion is to be made after the input data word of the current cycle. However, these are not limitations to the present invention, which may be practiced with data words of larger or smaller sizes, and with alternate conventions in denoting the data byte insertion position.

On page 8, please replace the paragraph beginning on line 14, with the following paragraph:

As illustrated in **Fig. 2a-2b**, input data word re-alignment unit **102a** for generating the intermediate data words comprises two portions, portion **102aa** and **102ab**. Portion **102aa** is employed to generate a pre-insertion data byte position of the current cycle (pre ins pos**[[t]]** cc), a post-insertion data byte position of the current cycle (post ins pos cc), a pre-insertion shift amount of the current cycle (pre ins shift amt cc) and a post-insertion shift amount of the current cycle (post ins shift amt cc); whereas portion **102ab** is employed to use the above described pointers and shift amounts of the current cycle to generate the two earlier described intermediate data words of the current cycle.

On page 10 and continually on to page 11, please replace the paragraph beginning on line 20, with the following paragraph:

As illustrated in **Fig. 3a-3c**, input data word alignment unit **102b** for generating the intermediate data words comprises three portions, portion **102ba**, portion **102bb**, and portion **102bc**. Portion **102ba**, similar to part of its counterpart, portion **102aa** of input data word alignment unit **102a**, is employed to generate a pre-insertion data byte position of the preceding cycle (pre ins pos**[[t]]** pc) and a post-insertion data byte position of the preceding cycle (post ins pos pc). Portion **102bb**, similar to the other part of its counterpart, portion **102aa** of input data word alignment unit **102a**, is employed to generate a pre-insertion shift amount of the preceding cycle (pre ins shift amt pc) and a post-insertion shift amount of the preceding cycle (post ins shift amt pc). Portion **102bc**, similar to its counterpart, portion **102ab** of input data word alignment unit **102a**, is employed to use the these pointers and shift amounts of the preceding cycle to generate the two earlier described intermediate data words of the preceding cycle.

On page 11, please replace the paragraph beginning on line 6, with the following paragraph:

Pre-insertion data byte position of the preceding cycle (pre ins pos**[[t]]** pc) points to the data byte position after which the insertion data bytes of the preceding cycle (if any) were made. Post-insertion data byte position of the preceding cycle (post ins pos pc) points to the data byte position at which the insertion data bytes of the preceding cycle ended. Pre-insertion shift amount of the preceding cycle (pre ins shift amt pc) denotes the amount of shifting (in units of data bytes) to be applied to the input data word of the preceding cycle to generate the intermediate data word having the pre-insertion data bytes of the input data word of the preceding cycle (if any) re-positioned appropriately. Post-insertion shift amount of the preceding cycle (post ins shift amt pc)

denotes the amount of shifting (in units of data bytes) to be applied to the input data word of the preceding cycle to generate the intermediate data word having the post-insertion data bytes of the input data word of the preceding cycle (if any) re-positioned appropriately.

On page 12, please replace the paragraph beginning on line 6, with the following paragraph:

Arithmetic operator **306** is employed to subtract a saved post insertion data byte position (saved post ins pos) from the data word size in units of data bytes (4'b1000 for the embodiment). Logical operator **308b** is employed to perform a bitwise OR logical operation on bits [4:3] of the saved post insertion data byte position (saved post ins pos). The result of the bitwise OR operation, is in turn used by selector **310b** to select either the difference of the above described subtraction operation performed by arithmetic operator **306** or zero (4'b0000), and output the selected value as the post-insertion data byte position of the preceding cycle (post ins pos pc). More specifically, selector **310b** selects the result of the above described arithmetic operation, and outputs the result as the post insertion data byte position of the preceding cycle, if the result of the bitwise OR operation is set. Otherwise, selector **310b** selects the zero value, and outputs as the prost insertion data byte position of the preceding cycle.